CLAIMS

What is claimed is:

1. A method for verifying a boot block program, the method comprising the acts of:

temporarily loading the boot block program into a first memory;

verifying the boot block program in the first memory; and

if verified, more permanently loading the boot block program into a second memory.

2. The method, as set forth in claim 1, wherein the act of temporarily loading comprises the act of temporarily loading the boot block program into a random access memory.

3. The method, as set forth in claim 1, wherein the act of temporarily loading comprises the act of temporarily loading the boot block program into a volatile memory.

4. The method, as set forth in claim 1, wherein the act of verifying comprises the act of authenticating the boot block program.

5. The method, as set forth in claim 1, wherein the act of verifying comprises the act of determining whether the boot block program is operable.

6. The method, as set forth in claim 1, wherein the act of more permanently loading comprises the act of semi-permanently loading the boot block program into the second memory.

7. The method, as set forth in claim 1, wherein the act of more permanently loading comprises the act of loading the boot block program into a read only memory.

8. The method, as set forth in claim 1, wherein the act of more permanently loading comprises the act of flashing the boot block program into a flash memory.

9. The method, as set forth in claim 1, wherein the acts are performed in the recited order.

10. A computer system comprising:

a host computer;

an appliance server operably coupled to the host computer, the appliance server having a storage memory and an execution memory;

a control operably coupled to the appliance server and to the storage memory to control storage of programs into the storage memory, the appliance server being adapted to signal the control to permit the appliance server to store a program in the storage memory; and

a security device operably coupled to the control, the security device being adapted to signal the control to permit the host computer to store a program in the storage memory.

11. The computer system, as set forth in claim 10, wherein the storage memory comprises a read only memory.

12. The computer system, as set forth in claim 10, wherein the storage memory comprises a non-volatile memory.

13. The computer system, as set forth in claim 10, wherein the storage memory comprises a flash memory.

14. The computer system, as set forth in claim 10, wherein the execution memory comprises a random access memory.

15. The computer system, as set forth in claim 10, wherein the execution memory comprises a volatile memory.

16. The computer system, as set forth in claim 10, wherein the security device comprises a switch.

17. A method of operating a computer system, the method comprising the acts of:

verifying a program of an appliance server; and

if not verified, signaling a host computer to load a replacement program into the appliance server.

18. The method, as set forth in claim 17, wherein the act of verifying comprises the act of authenticating the program.

19. The method, as set forth in claim 17, wherein the act of verifying comprises the act of determining whether the program is operable.

20. The method, as set forth in claim 17, wherein the act of signaling comprises the act of enabling a security switch.

21. The method, as set forth in claim 17, wherein the act of signaling comprises the act of determining whether a security switch has enabled the host computer to load the replacement program into the appliance server.

22. The method, as set forth in claim 17, wherein the program comprises a boot block program.

23. The method, as set forth in claim 17, wherein the program comprises a firmware program.

24. A method of operating a computer system, the method comprising the acts of:

during operation of a computer executing a first program in an execution memory and having a copy of the first program stored in a storage memory, replacing the first program stored in the storage memory with a second program by loading the second program into the storage memory;

verifying the second program stored in the storage memory; and

if not verified, reloading the first program from the execution memory into the storage memory.

25. The method, as set forth in claim 24, wherein the execution memory comprises a random access memory.

26. The method, as set forth in claim 24, wherein the execution memory comprises a volatile memory.

27. The method, as set forth in claim 24, wherein the storage memory comprises a read only memory.

28. The method, as set forth in claim 24, wherein the storage memory comprises a non-volatile memory.

29. The method, as set forth in claim 24, wherein the storage memory comprises a flash memory.

30. The method, as set forth in claim 24, wherein the act of verifying comprises the act of authenticating the second program.

31. The method, as set forth in claim 24, wherein the act of verifying comprises the act of determining whether the second program is operable.

32. The method, as set forth in claim 24, wherein the first and second programs comprise boot block programs.

33. The method, as set forth in claim 24, wherein the first and second programs comprise firmware programs.

34. A method of operating a computer system, the method comprising the acts of:

loading a program into a memory over a network connection;

if the network connection fails, re-establishing the network connection; and

once the network connection is re-established, continuing to load the program into the memory over the re-established network connection.

35. The method, as set forth in claim 34, wherein the act of loading comprises the act of flashing the program into a flash memory over the network connection.

36. The method, as set forth in claim 34, wherein the act of loading comprises the act of authenticating a user directing the loading of the program.

37. The method, as set forth in claim 36, wherein the act of continuing to load comprises the act of re-authenticating the user.